Service Fabric Customer Profile: Societe Generale and Qarnot Computing

Authored by Stéphane Bonniez from Societe Generale, Grégoire Sirou, Nicolas Duran and Erik Ferrand from Qarnot Computing in conjunction with Eric Grenon from Microsoft.

This article is part of a series about customers who've worked closely with Microsoft on Service Fabric over the last year. We look at why they chose Service Fabric and we take a closer look at the design of their application.

In this installment, we profile Societe Generale and Qarnot Computing, their grid computing application, and how they designed the architecture.

Societe Generale provides financial services to 31 million individuals and professionals worldwide, placing innovation and digital technology at the heart of its activities. Its corporate and investment banking business – SG CIB – offers global access to markets via solutions in equities, fixed income & currencies, commodities and alternative investments. SG CIB is committed to supporting corporates, financial institutions and investors by providing advisory, investment, and risk management solutions. The global markets platform draws from a wide pool of talent and expertise and is recognized for its worldwide leadership in equity derivatives and structured products and its leading positions in euro fixed income markets and cross-asset solutions.

Being a responsible company lies at the very heart of Societe Generale business lines' mission and its commitment to a greener and more inclusive economy has been accelerating since 2000.

Societe Generale is partnering with Qarnot Computing and Microsoft Azure to build its new financial simulation platform. Running market activities requires significant amounts of financial simulations running on large scale grid computing infrastructures. The new platform is flexible, scalable and environmentally responsible. It is cut out to support the growth of Societe Generale's business in an ever faster changing economy.

Founded in Paris in 2010, Qarnot Computing is a pioneer in distributed cloud and Smart Building technologies. They invented an innovative computing heater, the first of its kind, that uses the heat generated by the CPUs to heat buildings for free. Since 2014, more than 100 French households are heated with Qarnot Q.rads heaters. Their ingenuity has garnered several awards, including the 2015 Cloud Innovation World Cup Award.

Qarnot provides cloud computing through a distributed infrastructure where computing power is no longer deployed in concentrated datacenters, but spread throughout the city in the form of heaters and boilers. Their remote cloud computing powers private and public companies, including major banks, 3D animation studios, and research labs. But when a leading financial institution contacted Qarnot with a game-changing request for more compute power, Qarnot needed help from another cloud provider.

A Financial Simulation Platform

Financial simulations are computationally intensive: they typically involve several thousand calculation tasks, taking from a few seconds to several minutes each to compute. They also rely on a lot of data, sometimes hundreds of megabytes, such as the historical values of equity shares over several years. Still each task usually uses only a small portion of that data.

Simulation jobs are triggered by users at any time during working hours. Since Societe Generale has offices all around the world, that practically means at any time during the day, any day. Some of the simulations also have strong computation time constraints.

We designed a solution that:

- Exposes a simple REST API to client applications within Societe Generale.
- Handles calculation jobs ranging from a few tasks to several thousands.
- Provides caching of financial data for efficient dispatching of tasks.
- Scales with the number of jobs and tasks.
- Is available 24/7.

All this is achieved in a context where delivery must happen often (simulation libraries evolve continuously) and infrastructure costs must be kept as low as possible, although thousands of CPUs may be required to perform some simulations.

The key components of that solution are the following:

- A https web gateway exposing simulation services as a REST API.
- A collection of micro-services handling data caching and the orchestration of simulation jobs, from the dispatching of tasks to the retrieval of the results.
- Several grid computing providers. We currently target Azure Batch and Qarnot Computing's platform, but new providers can be added very easily.

The web gateway and the micro-services are native Service Fabric applications, all deployed in a scalable cluster in Azure Cloud.

Being able to target multiple cloud providers is a guarantee that a job will always find room to run at the best possible price.

"With Service Fabric, we were able to build a robust stateless microservice architecture in no time, giving us more time to focus our efforts on our product."

Nicolas Duran*, senior software engineer at Qarnot Computing*
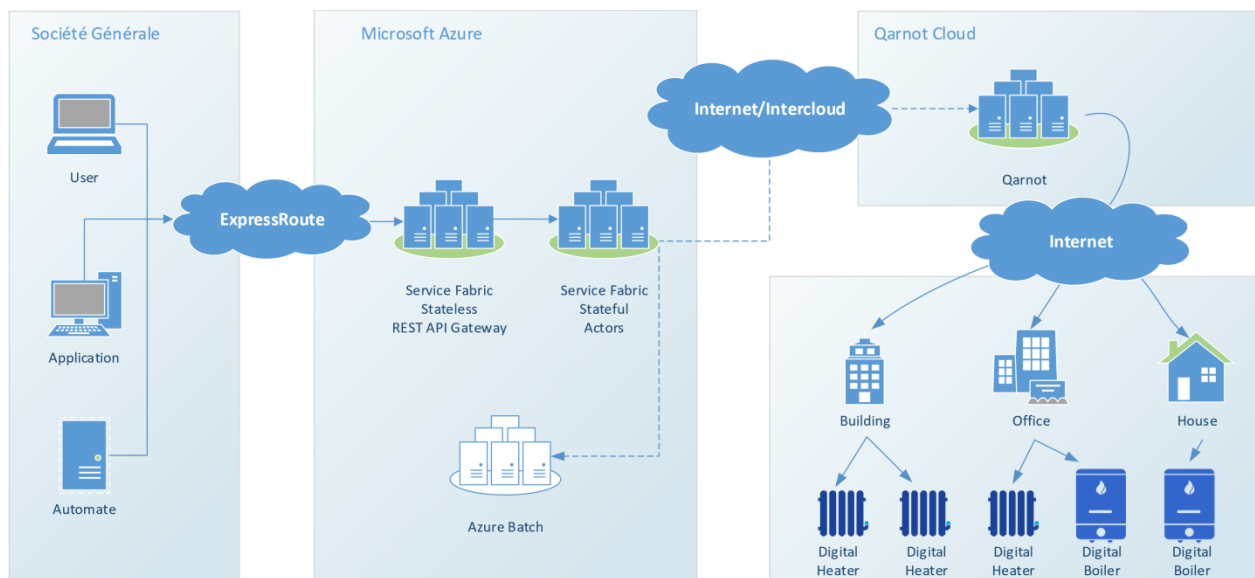
High-performance financial calculations are broken into discrete jobs and tasks by Service Fabric, then distributed to available cloud computing environments.

Service Fabric implementation

The Service Fabric part of the application is written in C#, with mix of services and actors, both stateless and stateful.

The cluster is a full-PaaS cluster in Azure Cloud, leaving the deployment and patching burden of the underlying software to Azure. The cluster nodes run on Windows.
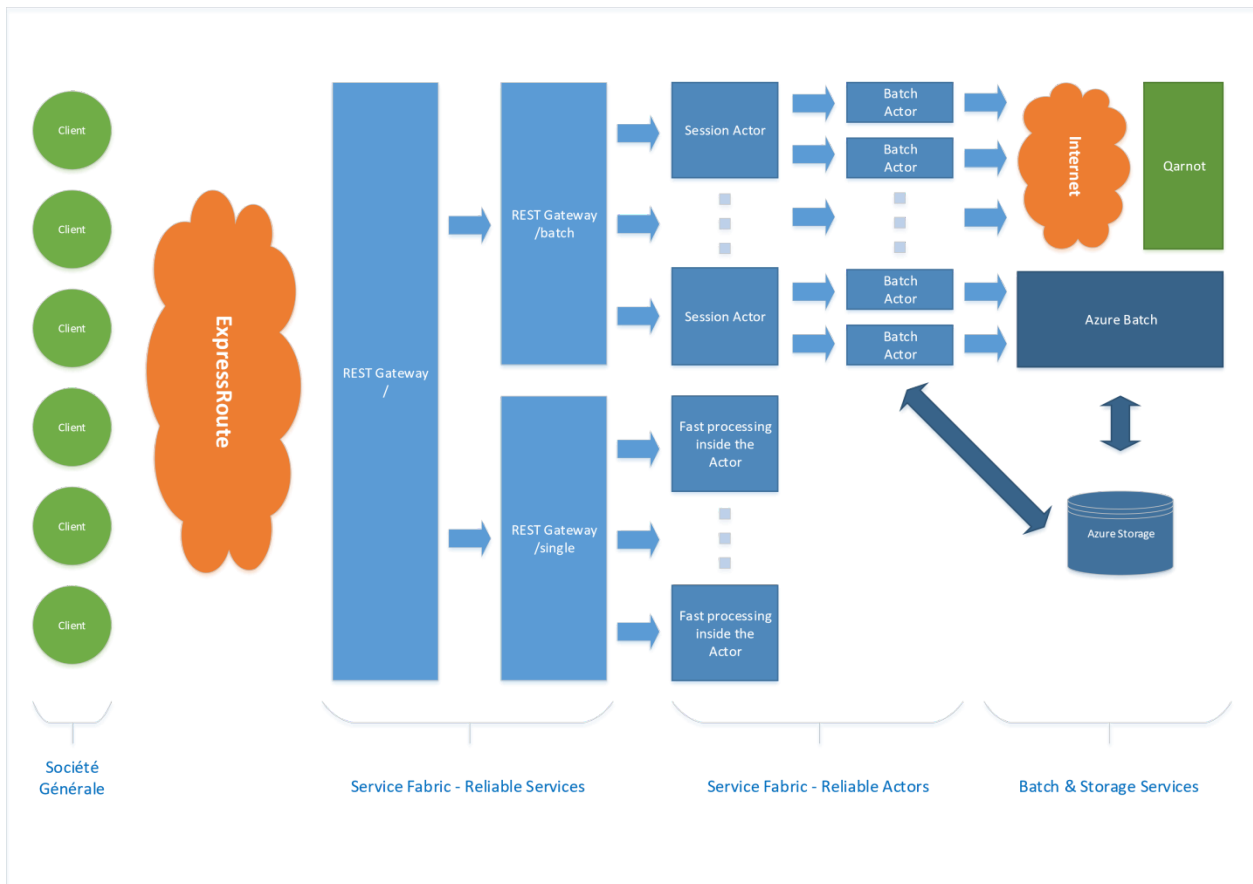
The web gateway is a **stateless reliable service**. Being the unique entry point of the application, it has to be highly scalable as multiple client applications within Societe Generale can run simulations concurrently. When the load rises, we can simply add new nodes to the cluster and Service Fabric will automatically launch more gateways and balance the load across the cluster.

Calculation jobs and tasks are implemented with **stateful reliable actors**. For instance, each task that is dispatched to the Azure Batch or Qarnot Computing platforms is "materialized" as an actor. Actors are really easy to write but they have very interesting properties:

- Their state is replicated on several instances across the cluster, so they are reliable.
- They are automatically distributed across the Service Fabric cluster, which provides scalability and load balancing.
- They are automatically unloaded from memory to disk if they have been inactive for some time and then automatically rehydrated in memory when called again. This saves memory and helps scale to more actors (so more simulation jobs and tasks).
- Their threading model guarantees that their state will always be consistent.

"With Service Fabric, developers can focus on business needs and rely on the platform for resiliency, load balancing or scalability. We can deliver better software, and do it faster."
Stéphane Bonniez, *Project Manager at Societe Generale*

Diagram labels (left to right): Client, Client, Client, Client, Client, Client / ExpressRoute / REST Gateway / / REST Gateway /batch / REST Gateway /single / Session Actor / Session Actor / Fast processing inside the Actor / Fast processing inside the Actor / Batch Actor / Batch Actor / Batch Actor / Batch Actor / Internet / Qarnot / Azure Batch / Azure Storage

Société Générale · Service Fabric - Reliable Services · Service Fabric - Reliable Actors · Batch & Storage Services

Advantages of Service Fabric

With a tight schedule, the joint Societe Generale and Qarnot team needed to ramp up fast. Service Fabric offered a complete toolset with its sophisticated runtime for building distributed microservices and its complete application management package for provisioning, deploying, monitoring, upgrading, and deleting deployed applications.

Given the deadline, the following Service Fabric benefits proved especially helpful:

- **Speed of development:** The powerful programming models provided by Service Fabric made it very easy for the developers to concentrate on business logic. Service Fabric managed the critical technical details—replication, resiliency, deployment systems, and more.

- **Self-healing**: The calculation solution required high resilience and availability. Service Fabric's ability to provide self-healing was a big benefit. For example, if a node or a process fails, the system automatically starts new instance.

- **Reliability**: As financial simulations involve a lot of calculation tasks depending on the same data, an easy optimization of the application was to hold a cache of these data. This way a client application can send all the data it will need then the tasks it wants to compute. Such a cache would be of poor value if it had to be rebuilt each time a node in the cloud is lost. Service Fabric makes it easy to write and manage reliable services. The developers used the Reliable Collections to handle data replication so it doesn't have to be coded at the application level. They simply specify how many times to replicate a state across nodes for reliability. In case of failure Service Fabric will just change the replica and the calculation will not lose data, avoiding relaunching the whole financial calculation.

- **Programming model:** Societe Generale and Qarnot took advantage of the productive programming models in Service Fabric to develop key components of their solution, from the gateway stateless service to the calculation task stateful service to the reliable actors used for task distribution.

- **Scalability:** Service Fabric provided the scale needed for the calculations, from one actor to thousands of actors. The developers saved countless hours—there was no need to manage the scale at the application level.

- **Application lifecycle:** The team can easily deploy a new version of the application with no downtime, or deploy multiple instances of the same application. The flexibility of the cloud and of Service Fabric development tools allowed us to fully integrate build, testing and deployment into Societe Generale's continuous integration pipeline: code is built and packaged, then tested on a local Service Fabric cluster and if all goes well it is automatically deployed to a development cluster in Azure. The same tests can run against the local cluster and the development cluster, which allows us to spot bugs very early in the chain. Once a version has been validated, it can be deployed to the production cluster the same way it was deployed to the development cluster and the same tests can be used to check that deployment went well.

"With Service Fabric, Societe Generale and Qarnot were able to speed up debugging and scaling thanks to the on-premises deployment and the perfect integration with the development tools."

Grégoire Sirou, *CTO of Qarnot Computing*

Summary

The challenge for Qarnot Computing and Societe Generale was to deliver a modular, scalable and resilient application in a very short timeframe.

Service Fabric was the right choice for the job. It let us concentrate on business logic, letting it do the mechanics. It's some sort of magic toolbox, offering powerful tools.

With Service Fabric, Qarnot Computing and Societe Generale have developed an innovative yet high-quality solution with just the right amount of effort.

Now that the simulation platform is in production, the team focuses on integrating new types of simulations and scaling the platform to handle them so that more and more client applications can move away from legacy systems. Future work will be centered on client management: the current platform lets clients use computational capacities as they want, future versions will integrate by-client capacity management and billing.